

**AMENDMENTS TO THE CLAIMS**

The listing of claims will replace all prior versions, and listings, of claims in the application:

**Listing of Claims:**

1. (original) A computer implemented method comprising:  
  
receiving requests for a shared resource, wherein access to the shared resource is controlled with a software lock;  
  
detecting that the software lock is unavailable;  
  
~~executing~~ emitting an operating system trace hook in response to the detecting, wherein the operating system trace hook is ~~adapted to record~~ records lock usage data; and  
  
analyzing the recorded lock usage data.
2. (original) The method of claim 1 wherein the lock usage data includes a timestamp corresponding to each request, a lock address corresponding to an address of the software lock, a stack call chain for each process making one of the requests at the time the corresponding request was made, and a request count.
3. (original) The method of claim 1 further comprising:  
  
receiving a lock release request from a process that currently holds the software lock;  
  
determining whether other processes are currently waiting for the software lock;  
  
emitting a second operating system trace hook and releasing the software lock in response to determining that other processes are currently waiting for the software lock, wherein the second operating system trace hook is also adapted to record lock usage data; and

releasing the software lock without emitting a trace hook in response to determining that other processes are not waiting for the software lock.

4. (original) The method of claim 1 further comprising:  
identifying one or more processes causing lock contention in response to the analysis of the lock usage data.
5. (original) The method of claim 4 further comprising:  
modifying the identified processes' usage of the shared resource.
6. (original) The method of claim 1 further comprising:  
detecting that the software lock is available in response to receiving the request;  
incrementing a lock counter that tracks the number of times the shared resource was requested; and  
setting ownership of the software lock to an identifier corresponding to an identifier of the requesting process.
7. (original) The method of claim 6 wherein the setting ownership further includes:  
requesting that a lock ownership field of the software lock be set to the requesting process identifier; and  
receiving a completion code indicating that the ownership has been set, wherein the incrementing of the lock counter is performed in parallel with the setting ownership by incrementing the lock counter after the request for the lock ownership field has been set and before the completion code has been received.
8. (currently amended) An information handling system comprising:  
one or more processors;

a memory accessible by the processors;

one or more shared resources accessible by the processors;

a nonvolatile storage device accessible by the processors;

one or more software locks, stored in the memory, for controlling access to the shared resources;

an operating system that controls a plurality of processes performed by the processors, the operating system including a trace hook facility; and

a software lock measurement tool for measuring lock contention, the software lock measurement tool including instructions that, when executed by at least one of the processors, cause the information handling system to perform steps comprising:

~~means for~~receiving requests from one or more of the processes for one of the shared resources, wherein access to the shared resource is controlled with one of the software locks;

~~means for~~detecting that the software lock is unavailable;

~~means for executing~~ emitting an operating system trace hook in response to the detecting, wherein the operating system trace hook is ~~adapted to record~~ records lock usage data; and

~~means for~~analyzing the recorded lock usage data.

9. (currently amended) The information handling system of claim 8 wherein the lock usage data includes a timestamp corresponding to each request, a lock address corresponding to an address of the software lock, a stack call chain for each process making one of the requests at the time the corresponding request was made, and a request count.

10. (currently amended) The information handling system of claim 8 wherein the software lock measurement tool includes further instructions that, when executed by at least one of the processors, cause the information handling system to perform further steps comprising:
- ~~means for~~ receiving a lock release request from a process that currently holds the software lock;
- ~~means for~~ determining whether other processes are currently waiting for the software lock;
- ~~means for~~ emitting a second operating system trace hook and releasing the software lock in response to determining that other processes are currently waiting for the software lock, wherein the second operating system trace hook is also adapted to record lock usage data; and
- ~~means for~~ releasing the software lock without emitting a trace hook in response to determining that other processes are not waiting for the software lock.
11. (currently amended) The information handling system of claim 8 wherein the software lock measurement tool includes further instructions that, when executed by at least one of the processors, cause the information handling system to perform further steps comprising:
- ~~means for~~ identifying one or more processes causing lock contention in response to the analysis of the lock usage data.
12. (currently amended) The information handling system of claim 8 wherein the software lock measurement tool includes further instructions that, when executed by at least one of the processors, cause the information handling system to perform further steps comprising:
- ~~means for~~ detecting that the software lock is available in response to receiving the request;

~~means for~~ incrementing a lock counter that tracks the number of times the shared resource was requested; and

~~means for~~ setting ownership of the software lock to an identifier corresponding to an identifier of the requesting process.

13. (currently amended) The information handling system of claim 12 wherein the step of ~~means for~~ setting ownership further includes:

~~means for~~ requesting that a lock ownership field of the software lock be set to the requesting process identifier; and

~~means for~~ receiving a completion code indicating that the ownership has been set, wherein the ~~means for~~ incrementing the lock counter is performed in parallel with the ~~means for~~ setting ownership by incrementing the lock counter after the request for the lock ownership field has been set and before the completion code has been received.

14. (currently amended) A computer program product stored on a computer ~~operable~~ storage media that includes instructions that, when executed by an information handling system, measure for measuring lock contention by performing steps comprising: ~~; said computer program product comprising:~~

~~means for~~ receiving requests for a shared resource, wherein access to the shared resource is controlled with a software lock;

~~means for~~ detecting that the software lock is unavailable;

~~means for~~ executing ~~emitting~~ an operating system trace hook in response to the detecting, wherein the operating system trace hook is ~~adapted to record~~ records lock usage data; and

~~means for~~ analyzing the recorded lock usage data.

15. (currently amended) The computer program product of claim 14 wherein the lock usage data includes a timestamp corresponding to each request, a lock address corresponding to an address of the software lock, a stack call chain for each process making one of the requests at the time the corresponding request was made, and a request count.
16. (currently amended) The computer program product of claim 14 wherein the instructions, when executed by an information handling system, perform further steps comprising:
- ~~means for~~ receiving a lock release request from a process that currently holds the software lock;
- ~~means for~~ determining whether other processes are currently waiting for the software lock;
- ~~means for~~ emitting a second operating system trace hook and releasing the software lock in response to determining that other processes are currently waiting for the software lock, wherein the second operating system trace hook is also adapted to record lock usage data; and
- ~~means for~~ releasing the software lock without emitting a trace hook in response to determining that other processes are not waiting for the software lock.
17. (currently amended) The computer program product of claim 14 wherein the instructions, when executed by an information handling system, perform further steps comprising:
- ~~means for~~ identifying one or more processes causing lock contention in response to the analysis of the lock usage data.

18. (currently amended) The computer program product of claim 17 wherein the instructions, when executed by an information handling system, perform further steps comprising:  
~~means for~~ modifying the identified processes' usage of the shared resource.
19. (currently amended) The computer program product of claim 14 wherein the instructions, when executed by an information handling system, perform further steps comprising:  
~~means for~~ detecting that the software lock is available in response to receiving the request;  
~~means for~~ incrementing a lock counter that tracks the number of times the shared resource was requested; and  
~~means for~~ setting ownership of the software lock to an identifier corresponding to an identifier of the requesting process.
20. (currently amended) The computer program product of claim 19 [[6]] wherein the step of ~~means for~~ setting ownership further includes steps comprising:  
~~means for~~ requesting that a lock ownership field of the software lock be set to the requesting process identifier; and  
~~means for~~ receiving a completion code indicating that the ownership has been set, wherein the ~~means for~~ incrementing the lock counter is performed in parallel with the ~~means for~~ setting ownership by incrementing the lock counter after the request for the lock ownership field has been set and before the completion code has been received.